

1. DAĻA

Šajā uzdevumā būs nepieciešams modelēt datubāzi. Ievērot.

- Neveidot vairāk tabulu un lauku kā nepieciešams uzdevuma izpildei.
- Tabulu, lauku nosaukumus un datu piemērus rakstīt latviski (izņēmums – datu tips). Nosaukumos nelietot latviešu valodas diakritiskās zīmes (piemēram, „āķis” vietā rakstīt „akis”) un atstarpes.
- Izmantot kādu no dotajiem relāciju tipu apzīmējumiem.

→	→	←
Viens pret daudziem (<i>one to many</i>). Izmantojot šo relāciju tipu, izvēlēties vajadzīgo virzienu.	Viens pret vienu (<i>one to one</i>).	Daudzi pret daudziem (<i>many to many</i>).

Relāciju tipi un to apzīmējumi

Analizēt piedāvāto problēmu un zemāk tabulās izplānot relāciju datubāzes struktūru (aizpildīt tikai tās tabulas un laukus, kuri ir nepieciešami).

- Izplānot atbilstošus laukus, kas atvieglo datu apstrādi. (3 punkti)
- Uzrakstīt atbilstošus datu tipus. (7 punkti)
- Ierakstīt vienu datu piemēru katrā no laukiem. (2 punkti)
- Sasaistīt tabulas ar atbilstošā tipa relācijām. (8 punkti)

Uzrakstīt, kāda datubāzes vadības sistēma tiks izmantota uzdevuma izpildei.

1. uzdevums (1 punkts). Izveidot vismaz vienu korektu lauku katrā no tabulām *Saimnieks un Produkti*.

2. uzdevums (1 punkts). Izveidot tabulās *Saimnieks un Produkti* visus nepieciešamos laukus.

3. uzdevums (1 punkts). Izveidot korektus laukus vismaz 3 tabulās.

4. uzdevums (1 punkts). Ievadīt vismaz vienu korektu datu tipu katrā no tabulām *Saimnieks un Produkti*.

5. uzdevums (1 punkts). Visiem nepieciešamajiem laukiem vismaz 2 tabulās ievadīt atbilstošus datu tipus.

6. uzdevums (1 punkts). Visiem nepieciešamajiem laukiem vismaz 3 tabulās ievadīt atbilstošus datu tipus.

7. uzdevums (2 punkti). Vismaz 2 tabulās ir ierakstu unikālās identifikācijas lauki, to norādīt pie datu tipa.

8. uzdevums (2 punkti). Vismaz 3 tabulās ir ierakstu unikālās identifikācijas lauki, to norādīt pie datu tipa.

9. uzdevums (2 punkti). Pamatot ar relācijām savienoto lauku tipu.

10. uzdevums (1 punkts). Izveidot relāciju starp 2 tabulām.

11. uzdevums (1 punkts). Izveidot relācijas starp 3 tabulām.

12. uzdevums (2 punkti). Izveidot atbilstošā tipa relāciju starp 2 tabulām.

13. uzdevums (2 punkti). Izveidot atbilstošo tipu relācijas starp 3 tabulām.

14. uzdevums (1 punkts). Ievadīt korektu testa datu piemēru vismaz 3 laukos.

15. uzdevums (1 punkts). Ievadīt korektu testa datu piemēru visos laukos.

Problēmas apraksts

Veikalam „Brūnaļa” ir nepieciešama datubāze (no savstarpēji saistītajām tabulām), kurā var ērti uzskaitīt no katra klienta (fiziskās personas) iepirktās ekoloģiskās produkcijas daudzumu un samaksāto naudu par iepirkumiem. Datubāzei jāsaturo:

- informācija par saimnieku, kurš palīdz ar viņu sadarboties;
- informācija par iepērkamajiem produktiem, t. sk. informācija, kad katrs produkts iegādāts, katra produkta cena par kilogramu vai grammiem vai litriem. Daži piemēri, kāda var būt iepirktā produkcija:
 - piens (govs 2 % 1,32 eiro/litrā vai 2,5 % 1,50 eiro/litrā, vai 3,5 % 1,80 eiro/litrā; kazas piens 3,60 eiro/litrā);
 - siers (govs svaigs siers 9 eiro kilogramā, svaigs ar ķīmenēm 9,50 eiro kilogramā, auksti kūpināts 13 eiro kilogramā; kazas svaigs 15 eiro kilogramā vai auksti kūpināts 17 eiro kilogramā);
 - krējums (svaigs govs krējums 7 eiro litrā vai skābais govs krējums 5 eiro litrā).

Vieta uzdevuma atbildei:

Izvēlīties un lietot kādu no relāciju tipiem (sasaistīt izvēlētos laukus, aizvelkot un novietojot vajadzīgo bultu):

Tabula saimnieks

Lauka nosaukums	Datu tips	Datu piemērs

Tabula produkti

Lauka nosaukums	Datu tips	Datu piemērs

Tabula

Lauka nosaukums	Datu tips	Datu piemērs

Tabula

Lauka nosaukums	Datu tips	Datu piemērs

2. DAĻA

PROBLĒMSITUĀCIJAS APRAKSTS

Atpūtas centram „Esi vesels” ir liels baseins ar 3 pirtīm: tvaika pirts, kurai temperatūra 40–50 °C un mitrums 70–100 %; sausā pirts jeb sauna, kurai temperatūra 90–110 °C un mitrums 10–20 %; mitrā pirts, kurai temperatūra 70–90 °C un mitrums 30–50 %. Katrā pirtī viedais skaitītājs saskaita apmeklētājus un viņu uzturēšanās ilgumu sekundēs un saglabā datus *.csv datnēs.

Atpūtas centrs regulāri atjauno statistiku, kas satur apmeklētāju raksturojošu informāciju, piemēram, par vecumu, apmeklēto pirti. Statistika par vairākiem gadiem ir pieejama savstarpēji nesaistītās *.csv datnēs (skatīt tabulu), līdz ar to informāciju ir grūti pārskatīt un analizēt.

Atpūtas centrs vēlas izveidot pārskatu ar dažādām datu atlasēs iespējām un vizualizācijām kādā no digitālajiem rīkiem, kas konkrētos datus attēlo lietotājam saprotamā un ērtā formātā, redzot katras dienas peļņu, kas palīdzētu regulāri vērtēt centra darbību un izveidot vēl vienu no visvairāk apmeklētajām pirtīm. Darbinieki labprāt sadarbosies ar izstrādātājiem.

Datums	Tvaika pirts	Sauna	Mitrā pirts
13.09.2022.	30	5	99
14.09.2022.	33	7	98
15.09.2022.	3	9	89
16.09.2022.	39	8	78
17.09.2022.	44	7	97
18.09.2022.	55	9	95
20.09.2022.	54	20	50
21.09.2022.	75	9	93

Kategorija	Cenas
Pensionāri	9 eiro
Pieaugušie no 10.00 līdz 15.00	15 % atlaide
Pieaugušie	12 eiro
Bērni no 8 līdz 16 gadiem	10 eiro
Bērni līdz 7 gadiem	par brīvu

Izpēte, mērķauditorija

1. uzdevums (2 punkti). Uzrakstīt vienu vispiemērotāko izpētes metodi, ar kuru var noskaidrot programmatūras produkta risinājumu, izmantojot dotos datus. Pamatot, kāpēc izvēlēta tieši šī izpētes metode.

Izpētes metode: _____

Pamatojums: _____

2. uzdevums (2 punkti). Definēt mērķauditoriju, kuru iesaistīs izpētes procesā, lai savā darbā izmantotu izstrādāto programmatūras produktu. Pamatot mērķauditorijas izvēli.

Mērķauditorija: _____

Pamatojums: _____

3. uzdevums (2 punkti). Sastādīt izpētes procesa plānu, uzrakstot un īsi paskaidrojot 1., 2., 3., 4. un 6. izpētes procesa soli. Plāna izveidē ņemt vērā 1. un 2. uzdevumā sniegtās atbildes.

3.1. _____

3.2. _____

3.3. _____

3.4. _____

3.5. Secināšana – izdaru secinājumus, lai varētu veikt turpmākās darbības.

3.6. _____

Programmatūras prasību specifikācija programmvadāmā risinājuma izstrādei

4. uzdevums (12 punkti). Ņemot vērā piedāvāto problēmas aprakstu par atpūtas centru, specifikācijā aprakstīt ievaddatus un izvaddatus, kurus lietos programmatūras produkta veidotājs.

4.1. Ievaddati, kas tiks doti. (1 punkts)

4.2. Ievaddatu tips, mērvienības, iespējamās vērtības. (2 punkti)

4.3. Veids, kādā tiks iesniegti ievaddati. (1 punkts)

4.4. Izvaddatu tips, mērvienības, iespējamie saglabāšanas veidi. (4 punkti)

Papildfunkcijas (kuras tu ieteiktu papildus problēmas aprakstā vai uzdevuma nosacījumos minētajām):

4.5. Piemēri ar konkrētiem ievaddatiem un atbilstošu sagaidāmo rezultātu. (4 punkti)

Veids, kā iegūt atbilstošu sagaidāmo rezultātu:

5. uzdevums (2 punkti). Kurš programmatūras izstrādes modelis ir piemērotākais konkrētā programmatūras produkta izstrādē?

Pamatot, kāpēc. Sasaistīt savu pamatojumu ar doto problēmu.

3. DAĻA

Mainīgo nosaukumus, kurus definē programmas kodā, rakstīt atbilstoši programmēšanas vides nosacījumiem (t. i., latviešu mīksto un garo burtu vietā izmantot atbilstošos latīņu burtus).

OOP 1

1. uzdevums (6 punkti)

Norādīt izvēlēto programmēšanas valodu. _____

1.1. Uzrakstīt programmas koda bloku, kas definē piemērotu klasi informācijai par transportlīdzekļiem CSDD datubāzē atbilstoši zemāk norādītajai informācijai, kā arī definēt šai klasei metodi, ar kuru izvada šādai klasei atbilstoša objekta īpašību vērtības. (4 punkti)

1.2. Izveidot vienu klases objektu ar zemāk minētiem testa datiem. (1 punkts)

1.3. Izmantot objekta metodi, lai izvadītu ekrānā visas definētās objekta īpašības. (1 punkts)

Apstrādājamā informācija par transportlīdzekli – zīmols, modelis, reģistrācijas datums CSDD datubāzē formātā dd.mm.gggg, pilna masa kilogramos (bez mērvienības), degvielas veida apzīmējums (iespējamie apzīmējumi: B – benzīns, BG – benzīns/gāze, D – dīzeļdegviela, E – elektriskais, BE – benzīna hibrīds, DE – dīzeļa hibrīds).

Testa dati:

zīmols: Audi

modelis: A4

reģistrācijas datums: 22.10.2019

pilna masa: 1800

degvielas veids: BG

OOP 2

2. uzdevums (32 punkti)

Norādīt izvēlēto programmēšanas valodu. _____

Noliktavā paredzēts uzstādīt iepakšanas robotu, kura uzdevums ir iepakot taisnstūra formas kastēs iespējami daudz no vienādiem kubveida klucīšiem veidotu dažādu taisnstūra paralēlskaldņa formas bloku. Kastes bloku iepakšanai tiek izvēlētas tā, ka to visi malu garumi ir proporcionāli klucīša izmēram (kastes jebkuras malas garums, kas dalīts ar klucīša malas garumu, ir vesels skaitlis). Klucīši var būt dažādās krāsās, bet bloks drīkst būt veidots tikai no vienādas krāsas klucīšiem. Viens bloks var būt veidots no viena, diviem, trīs vai četriem klucīšiem.

Programmētāju komandai dots uzdevums izveidot datorprogrammu minētā robota darbināšanai. Tev kā programmētāju komandas dalībniekam ir izvēlēta programmēšanas valodā jāsastāda kods, kas realizē tālāk minētos uzdevumus.

2.1. Definēt:

- klasi **kubs** (1 punkts), kurai ir īpašības (1 punkts):
 - o **malas garums** centimetros, vesels skaitlis intervālā no 2 līdz 10 ieskaitot;
 - o **krāsas nosaukums** (viens vārds) – teksts;
- metodes:
 - o **aprekinat_tilpumu**, kura aprēķina un atgriež veselu skaitli – kuba tilpumu kubikcentimetros (1 punkts);
 - o metode, kura likvidē objektu un ekrānā izvada paziņojumu, ka objekts likvidēts, paziņojumā norādot likvidētā objekta krāsu (1 punkts);
 - o metode, kura inicializē klases **kubs** īpašības (1 punkts).

2.2. Izveidot klases **kubs** konstruktoru, kas veic saņemto datu (argumentu) kontroli, izvadot atbilstošu paziņojumu (1 punkts) un uzstādot minimālo nosacījumiem atbilstošo vērtību (1 punkts), ja malas garums neatbilst nosacījumiem.

2.3. Izveidot jaunu klases **kubs** objektu **kubg**, kura krāsa ir zaļa un malas garums 10 centimetri (1 punkts).

2.4. Izveidot jaunu klases **kubs** objektu **kubr**, kura krāsa ir sarkana un malas garums 1 centimetrs (1 punkts).

2.5. Izvadīt ekrānā objekta **kubg** krāsu un tilpumu (1 punkts).

2.6. Izvadīt ekrānā objekta **kubr** malas garumu (1 punkts).

2.7. Dzēst objektu **kubr** (1 punkts).

2.8. Pārbaudīt, ka objekts **kubr** vairs nav pieejams, un izvadīt ekrānā atbilstošu paziņojumu (1 punkts).

2.9. Izvadīt ekrānā objekta **kubg** malas garumu (1 punkts).

2.10. Definēt klasi **bloks** (1 punkts), kura manto klasi **kubs** (1 punkts) un kurai ir:

- īpašības:
 - privātas (1 punkts):
 - o **kubu skaits** blokā, vesels skaitlis intervālā no 1 līdz 4 ieskaitot;
 - publiskas (1 punkts):
 - o **nosaukums**, kurš veidots no kuba krāsas un kubu skaita apvienojuma, piemēram, orange4;
 - o **forma**, vesels divciparu skaitlis ar iespējamām vērtībām 11, 12, 13, 14 un 22, kur pirmais cipars norāda vertikālā stāvoklī novietota bloka platumu (klucīšos), bet otrais – bloka augstumu (klucīšos);
 - o **derīgums** – vesels skaitlis, noklusētā vērtība 0, bet tiek uzstādīta uz 1, ja formas parametrs neatbilst noteikumiem;
- metode (1 punkts):
 - o **tilpums**, kura aprēķina un atgriež veselu skaitli – bloka tilpumu kubikcentimetros.

2.11. Izveidot klases **bloks** konstruktoru, kas veic saņemto datu (argumentu) kontroli, izvadot atbilstošu paziņojumu, ja:

2.11.1. parametra **forma** vērtība neatbilst nosacījumiem (1 punkts) un uzstāda objekta parametra **derīgums** vērtību uz 0, citādi uzstāda derīguma vērtību uz 1 (1 punkts);

2.11.2. kubu skaits blokā neatbilst nosacījumiem (1 punkts).

- 2.12. Izveidot klases **bloks** objektu, kas sastāv no 3 oranžas krāsas kubiem ar malas garumu 5 centimetri, formas numuru 13, un piešķirt tam nosacījumiem atbilstošu nosaukumu (4 punkti).
- 2.13. Izvadīt ekrānā izveidotā objekta nosaukumu un tilpumu (1 punkts).
- 2.14. Izveidot klases **bloks** objektu, kas sastāv no 5 zilās krāsas kubiem ar malas garumu 7 centimetri, formas numuru 23, un piešķirt tam nosacījumiem atbilstošu nosaukumu (2 punkti).
- 2.15. Izvadīt ekrānā izveidotā objekta nosaukumu un derīgumu (1 punkts).
- 2.16. Nomainīt objektam formas numuru uz 12 (1 punkts).
- 2.17. Izvadīt ekrānā izveidotā objekta nosaukumu un derīgumu (1 punkts).

4. DAĻA

Norādīt izvēlēto programmēšanas valodu. _____

1. uzdevums (13 punkti).

Izveidot API pieprasījumu norādītajai tīmekļa vietnei un atbilstoši uzdevuma nosacījumiem apstrādāt iegūtos datus.

Situācijas apraksts:

Dota Latvijas atvērto datu vietne – data.gov.lv. Jums tiek piedāvāta 2018. gada datu kopa „Atkritumu šķirošanas punkti”. Izveidot pieprasījumu uz doto resursu un apstrādāt tā atbildi, izvadot tikai tos atkritumu šķirošanas punktus, kuros var nodot kādu no šiem atkritumu veidiem: baterijas un akumulatorus vai nolietotās riepas, vai metālu. Izvadīt konkrētā punkta adresi un novada nosaukumu. Nodrošināt, ka programma analizē ārējā resursa atbildi, ja ārējais resurss neatbild, tad programmai jāizvada atbilstošs paziņojums. Ja ārējais resurss atbild ar tukšu atbildi, arī veikt atbilstoša teksta izvadi.

Pieprasījuma adrese uz šo resursu pieejama šeit:

https://data.gov.lv/dati/lv/api/3/action/datastore_search?resource_id=92ac6e57-c5a5-444e-aaca-ae90c120cc3d

Datu kolonnu atšifrējumi:

- „0 : Papīrs” – ir nododams papīrs.
- „1 : Plastmasa” – ir nododama plastmasa.
- „2 : Stikls” – ir nododams stikls.
- „3 : Metāls” – ir nododams metāls.
- „4 : Bioloģiski noārdāmie atkritumi” – ir nododami bioloģiski noārdāmie atkritumi.
- „5 : Tekstilmateriāli” – ir nododami tekstilmateriāli.
- „6 : Elektriskās un elektroniskās iekārtas” – ir nododamas elektriskās un elektroniskās iekārtas.
- „7 : Apgaismes iekārtas un spuldzes” – ir nododamas apgaismes iekārtas un spuldzes.
- „8 : Baterijas un akumulatori” – ir nododamas baterijas un akumulatori.
- „9 : Sadzīvē radušies bīstamie atkritumi” – ir nododami sadzīvē radušies bīstamie atkritumi.
- „10 : Nolietotās riepas” – ir nododamas nolietotās riepas.

Burts „x” norāda, ka konkrētajā punktā ir iespējas nodot attiecīgos atkritumus.

- 1.1. Izveidot izsaukumu uz doto resursu (2 punkti).
- 1.2. Analizēt ārējā resursa atbildi, vai tas atbild ar korektu atbildi (2 punkti).
- 1.3. Izvadīt atbilstošu tekstu, ja serveris neatbild (2 punkti).
- 1.4. Izvadīt atbilstošu tekstu, ja serveris atbild ar tukšu (tukšu masīvu) atbildi (1 punkts).
- 1.5. Izvadīt attiecīgā atkritumu nodošanas punkta adresi, kurā var nodot baterijas un akumulatorus (1 punkts).
- 1.6. Izvadīt attiecīgā atkritumu nodošanas punkta novada nosaukumu, kurā var nodot baterijas un akumulatorus (1 punkts).
- 1.7. Izvadīt attiecīgā atkritumu nodošanas punkta adresi, kurā var nodot nolietotās riepas (1 punkts).
- 1.8. Izvadīt attiecīgā atkritumu nodošanas punkta novada nosaukumu, kurā var nodot nolietotās riepas (1 punkts).
- 1.9. Izvadīt attiecīgā atkritumu nodošanas punkta adresi, kurā var nodot metālu (1 punkts).
- 1.10. Izvadīt attiecīgā atkritumu nodošanas punkta novada nosaukumu, kurā var nodot metālu (1 punkts).

Norādīt izvēlēto programmēšanas valodu. _____

2. uzdevums (15 punkti).

Izveidot latviešu vārdu sarakstu alfabētā secībā no ievadītajiem vārdiem

Uzdevuma apraksts

Jums jāizveido programma, kurai, ievadot vārdu, tas tiek novietots datu struktūrā atbilstoši latviešu alfabētam. Vietu alfabētā nosaka ievadītā vārda pirmais burts.

Ja vārda pirmais burts atkārtojas, proti, tiek ievadīts vārds, kura vieta sarakstā jau ir aizņemta, vārdu apmaina pret ievadīto. Pārbaudīt lietotāja ievadi, lai lietotājs nevarētu ievadīt nederīgas vērtības. Par derīgu vērtību tiek uzskatīts tikai viens latviešu vārds, kurš sākas ar lielo burtu. Programmai jāpaziņo, ja ievadītā vērtība neatbilst minētajiem nosacījumiem, tad programmai jāpieprasa ievadīt jaunu vērtību. Ievadot vārdu, ir jāizvada programmas darbības solī, piemēram, ja vārds vēl nav ievietots, programma izvada teikumu „Pievienoju vārdu 1. vietā”. Ņemt vērā, ka vārdu uzskaitē alfabētā sākas ar kārtas skaitļa vietu Nr. 1.

Programma darbojas, līdz tiek pilnībā aizpildīts viss alfabēta saraksts.

Lai pārbaudītu programmas darbību, varat izmantot doto vārdu sarakstu:

Ainaži, Saulkrasti, Dobeles, Sigulda, Tukums, Liepāja, Talsi, Ludza, Cēsis, Gulbene, Ventspils, Vecumnieki, Engure, Ērgļi, Staicele, Kuldīga, Aizpute, Krāslava, Madona, Jūrmala, Rīga.

Punkts par katru no šādām programmas darbībām:

Vārda ievads

2.1. Nodrošināt vārda ievadīšanu.

2.2. Pārbaudīt lietotāja ievadīto vērtību, vai tas ir vārds (visas citas vērtības netiek ieskaitītas kā pareizas).

2.3. Katru reizi ievadīt tikai vienu vārdu, nevis vārdu virkni.

2.4. Pārbaudīt, vai vārda pirmais burts ir lielais burts.

2.5. Atkārtoti ievadīt vērtību, ja ievadītā vērtība bijusi kļūdaina.

Datu struktūra

2.6. Izveidot datu struktūru, kurā uzglabāt latviešu alfabētu un tam atbilstošā sākumburta pozīcijas kārtas numuru.

2.7. Izveidot datu struktūru, kurā glabāt jauno izveidoto sarakstu.

2.8. Noteikt ievadītā vārda pirmo burtu.

2.9. Ja vārda pozīcija sarakstā jau ir aizņemta ar kādu vārdu, tad to aizvietot ar ievadīto vārdu.

2.10. Ja vārda pozīcijas sarakstā vārda nav, tad tajā tiek ievietots ievadītais vārds.

2.11. Pārbaudītais vārds tiek ievietots alfabēta secības atbilstošajā pozīcijā, ņemot vērā tā pirmo burtu.

2.12. Izdrukāt katru programmas darbību kā lasāmu teikumu.

2.13. Izvadīt kļūdas paziņojumu, ja ievadītais vārds neatbilst noteikumiem.

2.14. Datu struktūrā nedrīkst būt nultais elements.

2.15. Programma darbojas, līdz viss saraksts ir aizpildīts.